

QUANTA

www.quanta.org.uk

CONTENTS

EDITORIAL.....	John Gilpin.....	3
COPY DATE for next issue.....		3
CHAIRMAN'S NOTES.....	John Mason.....	4
COMMITTEE PRECIS.....	Sarah Gilpin.....	5
AGM 2008.....	Sarah Gilpin.....	5
QUANTA POSTER.....		6
HOW GPS RECEIVERS		
FIND THE TIME.....	Hugh Rooms.....	7
SOLIHULL WORKSHOP UPDATE	Simon Goodwin.....	9
FOR SALE AND WANTED.....	John Gilpin.....	11
CHRISTMAS CHALLENGE.....	John Gilpin.....	12
SNAKE.....	Steve Poole.....	12
TRADER'S CORNER.....		17
QUANTA NEWS.....	John Gilpin.....	19
POLY GONEMAD	Steve Poole.....	21
PRIORITIES.....	Steve Poole.....	22
REPEAT After Me.....	Steve Poole.....	24
STADDON.....	Steve Poole.....	25
TIC TAC TOE.....	Steve Poole.....	30
SUBGROUP MEETINGS.....		33

Volume 24. Issue 6.
Dec 2007/Jan 2008

Information on the Association

Membership of QUANTA, the independent QL user group, is by annual subscription. The Membership Secretary can supply full details. Copies of the association's constitution & annual accounts are available from the Secretary. There is an extensive software library. There are active national subgroups. Details are given in the Member's Guide and in this magazine.

QUANTA Committee - 2007/2008

Chairman John Mason 73 Chewton Common Road, Highcliffe, Christchurch, BH23 5LZ
Email: chairman@quanta.org.uk Tel: 01425 275894

Secretary Sarah Gilpin 181 Urmston Lane, Stretford, Manchester. M32 9EH
Email: secretary@quanta.org.uk Tel: 0161 865 2872

Treasurer ,
Membership Secretary
& Missed Issues John Gilpin 181 Urmston Lane, Stretford, Manchester. M32 9EH
Email: treasurer@quanta.org.uk Tel: 0161 865 2872
Email: membership@quanta.org.uk

Magazine Editor

Email: editor@quanta.org.uk

Librarian

Email: librarian@quanta.org.uk

Helpline

Email: helpline@quanta.org.uk

Web Master Dan Abbott All contact with Web Master by email please.
Email: webmaster@quanta.org.uk

Back Issues Requests Basil Lee 31 Fairdale Gardens, London. SW15 6JW
Email: backissues@quanta.org.uk Tel: 0208 789 1976

QUANTA reserves the right to publish, reject or edit any material submitted. The opinions expressed are solely those of the contributors. Member's private small ads - free (up to 50 words). Non-members private small ads (and members after 50 words) 50p per 10 words or part thereof. Terms strictly paid up-front with order. Receipt if requested. All copyrights and trademarks are hereby acknowledged.

As this is the last issue of QUANTA Magazine for this year it seems appropriate for me to put a few words into the editorial.

It pleases me greatly to see and/or hear comments regarding the improvements that we (the editorial team - and it is a team) have made to the Magazine. No matter how good a team we are, publishing this Magazine would not be possible without our contributors. This year, a total of fourteen different contributors (including the editorial team) have submitted articles for publication and only one contributor has provided copy for all this year's issues. Thank you Steve Poole. Others have provided considerable contributions particularly Hugh Rooms (GPS on the QL) and Sarah Gilpin (Committee Precis and other matters which have kept us up to date with the way things are going with Quanta). Other regulars have been Dilwyn Jones, George Gwilt, David Buckley, Dan Abbott, Geoff Wicks, Tony Firshman, Albert Newell, Mike Bedford-White and John Roberts. (apologies to anybody that I have left out). Thank you all very much for your contributions and I hope that you (and perhaps some others) will provide us with copy in the coming year.

COPY DATE for FEBRUARY/MARCH 2008

ISSUE is:

5TH FEBRUARY 2008

Please ensure that your copy reaches the editor on or before that date. Copy on Paper or Magnetic Media or by Email.

All Contributions are Welcome.

Copy on Microdrive or Floppy Disk or CD to be accompanied by hard copy printout.

Copy by Email to - editor@quanta.org.uk

This Magazine and all the articles etc. within it are copyright © QUANTA 2007/8 unless otherwise stated. Reproduction and redistribution in any form is not allowed without the express written permission of the publisher. All copyrights and trademarks are hereby acknowledged.

Quanta Magazine

Once again, Steve Poole has provided us with the opportunity to challenge you all over the forthcoming festive season. See elsewhere in this issue for details and let's hope that one or two of you accept the challenge this year.

Thank you to everyone who has assisted us in producing this Magazine and may I take this opportunity to wish all our membership (and others) a Very Merry Christmas and a Happy and Prosperous New Year 2008.

John Gilpin
(Joint Editor)

CHAIRMAN'S NOTES

John Mason

In the last issue of the magazine I highlighted the need for members to come forward and fill vacant Committee places.

To my great concern I have yet to receive any response to my challenge.

The need for Committee members to carry out the duties of Librarian/Software Controller, Helpline, and Magazine Editor continues to be unheeded. This places the present Committee in an invidious position.

I have to remind you that nominations for the 2008 Committee need to reach the Secretary by or before 1st February 2008.

Will you help to keep QUANTA going?

I wish ALL Members a Very Merry Christmas and a Happy New Year

John Mason
(Chairman & Joint Editor)

There have been no committee meeting's since the Workshop in Solihull. The committee is currently awaiting firm quotations from three venues in Solihull for weekends in April 2009. It is hoped that we shall find one suitable to hold the 2009 AGM and any accompanying celebrations for QL is 25.

Do not forget that we are looking for new members for the committee and nomination papers were sent to you all with the last magazine. Please remember that nominees and their nominators must be current members of QUANTA.

All nominations should be returned to me no later than February 1st. 2008. - Thank you.

AGM 2008

Sarah Gilpin

There was only one offer, from Nemqlug, to hold the AGM in 2008. Therefore it has been agreed that the AGM 2008 will be held in Manchester.

The venue will be the "Endeavour", 3rd Davyhulme Scout Headquarters, Conway Road, Davyhulme, as in previous years. The date is Saturday April 12th and Sunday 13th 2008. Nemqlug will host a dinner at the Pond Quay on the Saturday night. Further information will be in the next magazine regarding local accommodation etc.

If anyone has any specific questions to ask at this early stage, please address them to: secretary@quanta.org.uk



**A Very Merry Christmas
and a Happy New Year
to you all.**

Sarah G.

Independent QL Users Group

World-wide Membership is by subscription only,

Offering the following benefits:

Bimonthly Newsletter – up to 40 pages

Massive Software Library – All Free!

Free Helpline and Workshops

Regional Sub-Groups. One near you?

Advice on Software and Hardware problems

Subscription just £14 for UK members

Overseas subscription £17

Barclaycard: Visa: Access: MasterCard: Accepted

Now in our Twenty Fourth Year

Further details from the Membership Secretary

**John Gilpin, 181, Urmston Lane
Stretford, Manchester, M32 9EH (UK).
Tel. +44 (0) 161 865 2872**

Or

Visit the Quanta Web Site

<http://www.quanta.org.uk>

E-mail: membership@quanta.org.uk

Next QUANTA Sponsored Event

NEMQLUG (North East Manchester QL User Group)

Presents

MANCHESTER WORKSHOP & QUANTA AGM

3rd Davyhulme Scout Headquarters,

“The Endeavour”, Conway Road,

Davyhulme,

Manchester. M41 7FJ

Saturday & Sunday 12th /13th April 2008

From 10.00 am. to 4 pm. daily. AGM at 2.00 pm. Sunday.

Ring Sarah Gilpin on
0161 – 865 2872 for full details.

Quanta Magazine

[Following the completion of a very interesting Article by Hugh Rooms on GPS on the QL, I have received this article with a little more detail to satisfy those of you who requested it. Hugh is quite happy to correspond with anybody regarding the finer points of GPS work and the mathematics involved - and other topics too. Please contact editor@quanta.org.uk for Hugh's contact details - Ed]

HOW GPS RECEIVERS FIND THE TIME **Hugh Rooms**

In articles already published here I described the Global Positioning System and how it is used to find the accurate location of any place on earth. At the time I wrote them I did not understand how the relatively cheap clock in the receiver is synchronised with the atomic clocks in the satellites, which is really at the heart of the way it determines its position. Since then I've found that out, and the result is here.

A summary and reminder: GPS is a method of location using the calculated distances from a swarm of about two dozen artificial satellites, orbiting at around 12000 miles from the centre of the earth in twelve hour orbits, broadcasting data of time and their positions, on about 1.5 GHz, that allow a suitable receiver to calculate its own position in latitude and longitude, and its height above a datum. They circulate along paths closely monitored and corrected by ground stations, and they have, of course, to be controlled to a greater accuracy than the position you are trying to calculate. The orbits criss-cross in the sky so that any place on earth always has enough in view to get a good fix.

By measuring the times the radio signals take to arrive, a receiver is able to calculate its distances from them. Precisely each second every satellite sends its own unique signal; the delay in receiving it is the time taken for the signal to travel, multiply by the speed of propagation, and you have the distance. The signal, whizzing along at around 300,000 kilometres per second or 30 cm every nanosecond, takes about 68 milliseconds to arrive for a satellite overhead, and about 86 milliseconds for one on the horizon.

This obviously needs a very precise knowledge of the time the radio signal set off, so each satellite has a number of atomic clocks on board, again monitored and controlled from the ground so that they all synchronised within the satellite and throughout the swarm. The speeds and precision involved require that Einstein's Relativity is taken into

Quanta Magazine

account, so equipment to compensate for it was installed in the satellites; but some engineers doubted the need so it wasn't turned on at first. In the first three weeks of the trial the errors accumulated at the equivalent of 10km a day.

I won't go into this any more, but a readable account with useful further references is at:

http://en.wikipedia.org/wiki/Sagnac_effect.

It is difficult to make a correspondingly accurate clock reasonably cheaply for the receiver so it must synchronise its own quartz crystal controlled clock to the GPS time, as I will describe in a moment. A pay-off from this is that you can get a very accurate time check -- potentially many times more accurate than the MSF transmissions used in radio-controlled clocks, which, with a 60kHz carrier, can be kept correct only to about one millisecond.

To get a fix, four satellites are needed: the distance from the first defines a spherical 'surface' on which the receiver (strictly its antenna) must be, somewhere. The same applies to a second satellite, so the antenna is now known to be located on the circle where this 'sphere' cuts the first, since it must be on both. A third sphere cuts this circle in two places, and a fourth sphere should pass through just one of these points, giving a unique position relative to the satellites' frame of reference. With the receiver's clock unadjusted, this fourth sphere will miss the point established by the first three, but adding or subtracting a SAME small correction to EACH measurement will always make them coincide. Applying this correction to the ground receiver's clock synchronises it with satellite time, and a succession of these corrections will also give the rate of gain or loss in the receiver's clock.

However these measurements and calculations are not precise, so, in subsequent fixes, the 'spheres' will not meet exactly at the same point. More satellites' distances are measured giving a spread of intersections over a small space called a "resection". A calculation from the points defining the resection gives the most probable position (MPP in the jargon) and an average correction for the clock, and its volume is a measure of the additional inaccuracy of the fix, called "Dilution Of Position" or DOP, above that expected from the inherent tolerances of the system itself. It is a remarkably clever and ingenious system.

Quanta Magazine

A more detailed explanation is in: Dan Kalman, "An Undetermined Linear System for GPS," College Mathematics Journal, Vol. 33 (2002), pp. 384-390. He describes a method, with examples, of the maths that could be done to solve the equations involved to obtain the time and position from the GPS signals. Although, because of practical considerations, this is not the way it is actually done, I found it really enlightening and an excellent first step to a full understanding of the subject (not that I've achieved that yet!) I downloaded it from:

<http://www.american.edu/academic.depts/cas/mathstat/People/kalman/#MathLink>

[Hugh also reminds us that a start towards the mathematics of GPS can be found at

www.colorado.edu/geography/gcraft/notes/gps.gps.html

Peter H Dana, The Geographer's Craft Project, Department of Geography, The University of Colorado at Boulder. A detailed but rather technical overview of the GPS system by, I believe, one of the designers which does give the formulae for doing the calculations, with examples, but no derivation or explanation.

SOLIHULL WORKSHOP UPDATE Simon Goodwin

[Following the brief report of the QUANTA Workshop held in Solihull on Saturday 6th October 2007 in the last issue of Quanta Magazine, I received this email from Simon Goodwin regarding his presentation - Ed]

H i Quanta,

Here's the picture of the audience taken by SuperBASIC at my presentation at the Birmingham workshop. The QL camera-control program DIGICAM_BAS was running on UQLX on the Knoppix Gnu Linux system I gave away on the day, controlling a Kodak DC200+ digital camera via a 5600 baud serial interface. All the software used was free.

I also demonstrated Al Boehm's General MIDI player, again running in Minerva SuperBASIC on UQLX, controlling a Casio CZ-230S bought for the DIY MIDI project by NESQLUG several years ago, using Linux /dev/midi

Quanta Magazine

rather than my DIY MIDI QL hardware project, and QLSSS sampled sound replay via an Ensoniq PCI sound card routed by UQLX to the standard Linux /dev/dsp - as well as very high resolution Qdos-compatible displays.

All these demonstrate the compatibility of UQLX and its ability to take advantage both of Qdos-based drivers and modern hardware in a free open-source system. I also showed the latest progress of the QL2K emulator - written for Windows but running smoothly on Linux, thanks to WINE - and the limitations of QPC in the same configuration, which stem from its specific assumptions about the underlying system.

Cheers,

Simon N Goodwin

simon@mooli.org.uk



Wanted:

External disk drive units (or just cases) - single or dual units. There must be plenty out there from people who have upgraded.

For Sale:

- 10 - Epson Stylus Colour 850 printers
- 2 - Epson Stylus 900 printers
- HP Deskjet 660C Inkjet printers

All **ONLY** £40.00 each plus £15.00 post and packing

Contact: Rich Mellor, 3, Dale View Court, Fulford, Stoke-on-Trent, Staffordshire. ST11 9BA or by email at:

rich@rwapservices.co.uk

Wanted:

QUANTA are always on the lookout for Second User QL related items which are no longer required by their owner.

Do you know where there is some hardware, Software (original) books etc laying about doing nothing? Give me a call or send me an email.

John Gilpin QUANTA Treasurer. Contact details inside front cover.

For Sale:

QUANTA have the following second user Items for sale:

Reformatted Microdrive Cartridges - 4 in Plastic Wallet - £0.30
(Last few remaining - First Come, First Served)

And

Reformatted 3.5" DD Floppy diskettes - £0.20 ea - 10 for £1.50
100 for £11.50

Quanta Magazine

Jan Jones' Book "QL SuperBASIC - The Definitive Handbook"

This is the book which no QLer can afford to be without and the one which QUANTA did a Limited Reprint Edition in 1989 after the original print run sold out so quickly.

A Real Bargain at £1.00 each. (Last few remaining)

All the above items are subject to post and packing charges at cost.

Enquiries to treasurer@quanta.org.uk

CHRISTMAS CHALLENGE 2007

John Gilpin

Incorporating Software by Steve Poole.

Once again, Steve Poole has given us the software around which I have based a programming challenge to test your mental abilities after consuming masses of Christmas food & drink. Firstly, Steve's comments:

SNAKE

Steve Poole

Back in 1984, a colleague of my wife's was lying on an Atlantic beach here in France, when an idea for a computer game suddenly dawned on him. So he went straight back to his seaside hotel and plugged in his Oric Atmos. A few days later he had finished coding and took his work to a computer club meeting and ran the program for people to see. It aroused enormous interest immediately, and he was offered money for copies of his code.

In life he was a Mathematics teacher, but his program had been seen by a telecommunications expert who recognised that the method used also had applications in packet-switching: Finding routes through a web allowing strings of data-packets to follow the same paths. He was advised to take sabbatical leave from the Education Ministry and get a contract in telephone research. In fact, the telecoms authorities realised that the program could be sold in tens of millions as a game, and consequently offered him immediate retirement, or a permanent research post. He was tempted by the offer, and so gave up 'his' author's rights to them.

Quanta Magazine

I never saw the guy again, but the game he wrote was THE Magic Snake, one of computing's all-time greats. Nowadays the program is usually given away with PCs or mobile phones, and by now must have been sold in hundreds of millions! So I decided that it was time to write a version for the QL, with new features so as not to infringe on copyright.

After some thought I settled on a way of route-finding through a grid, linked to queue-handling. My solution was to use two arrays. The first would hold the empty grid, then the coordinates of the obstacles to avoid, 'food' objects to absorb and finally the snake's body. The rectangular grid allows navigation using arrow-keys for the snake's 'head'. Each array-index 'cell' number is calculated from its coordinates, and then the cell is inspected to see if it already contains an object which needs to be treated accordingly. If the spot is empty, a new head is drawn and put in the matrix. Otherwise you crash!

The second array holds the list of coordinates too, and uses three counters, one to record each 'head' cell as it is reached, a second to record the current 'tail' position, and a third to see if all the food-objects have been eaten. So all the program has to do is test for an arrow-key, test for an obstacle and erase the last tail-cell. If no key has been pressed, the snake keeps going in the same direction.

Programming this caused me several hours of anguish, as the tail-erasing process refused to operate as expected, but I could find no error in the logic. Eventually I found spurious behaviour whilst debugging... until I reset the PC. Then the code ran as expected, so I finally put this episode down to PC bugs. Without this problem, the program would only have taken about four hours to write, test and debug.

As usual I wanted to include levels of difficulty, so you can choose the size of grid and the reaction-speed. Indeed, it should be equally fast on any machine, as it uses INKEY\$ delays to make it fully compatible. I then modified the code so that initial obstacles cannot generate unavoidable dead-ends. This involves using simplified maze-construction routines... Obstacles are 'X', food is 'O' and the snake's body is 'S'. Remember, if you go off the edges you lose too...

The maximum grid size is 9 by 9 to simplify the initialisation code. For this year's **QUANTA CHALLENGE**, try adapting the program to use a 22 by 22 grid in CSIZE 0,0 to make the game more complicated. This mainly adapts the initialisation, as the main code is independent of dimension. But

Quanta Magazine

just play around with the delays, quantity of obstacles and fruit as much as you like to tailor the difficulty to your own personal taste. Remember, this code was written rapidly and can certainly be greatly optimised and improved... The randomness of the obstacles means that a 'winning' strategy is less obvious than first appears to be the case.

Happy Swerving!

```
100 ::
110 REMark Snake_bas by S.Poole, v7feb2007
120 REMark for Quanta
130 :
140 CLEAR: OPEN#1,con_16: WINDOW#1,512,256,0,0
150 PAPER 0: INK 7: CSIZE 3,1: CLS
160 pr 1,1,'Difficulty (6 to 9)?'
170 i$=INKEY$(#1,-1): IF i$="": GO TO 150
180 IF i$ INSTR '6789': n=i$: n1=n-1: ELSE GO TO 150
190 SElect n:=6:wt=125:=7:wt=100:=8:wt=75:=9:wt=50
200 CLS: PAPER 2: DIM g(512,3),q(512,2)
210 z=0: o=1: s=2: x=3: ct=1: kt=1: cnt=0: try=0
220 :
230 REMark Draw the grid:
240 FOR f=1 TO n
250   FOR j=1 TO n: pr j,f,' '
260 END FOR f
270 :
280 REMark Get random obstacle positions:
290 FOR f=1 TO n+n/3
300   ac=RND(2 TO n1): dn=RND(2 TO n1): p=(dn-1)*n+ac
310   IF g(p,3)=x: GO TO 300
320   :
330   REMark Eliminate obstacles on a diagonal:
340   b2=(dn-1+1)*n+ac+1: IF g(b2,3)=x: GO TO 300
350   b4=(dn-1-1)*n+ac+1: IF g(b4,3)=x: GO TO 300
360   b6=(dn-1-1)*n+ac-1: IF g(b6,3)=x: GO TO 300
370   b8=(dn-1+1)*n+ac-1: IF g(b8,3)=x: GO TO 300
380   :
390   g(p,1)=ac: g(p,2)=dn: g(p,3)=x: pr dn,ac,'X'
400 END FOR f: INK 1
410 :
420 REMark Get random snake food:
430 rd_n=RND(n*3 TO n*4)
440 FOR f=1 TO rd_n
450   ac=RND(1 TO n): dn=RND(1 TO n): p=(dn-1)*n+ac
460   IF g(p,1)=ac AND g(p,2)=dn: GO TO 450
```

Quanta Magazine

```
470  g(p,1)=ac: g(p,2)=dn: g(p,3)=o: pr dn,ac,'O'
480  END FOR f: INK 4
490  :
500  REMark Draw snake head:
510  ac=RND(1 TO n): dn=RND(1 TO n): p=(dn-1)*n+ac
520  IF g(p,1)=ac AND g(p,2)=dn: GO TO 510
530  pr dn,ac,'S': q(1,1)=ac: q(1,2)=dn
540  :
550  REMark Main game:
560  left=192: right=200: up=208: down=216: rn=RND(1 TO 4)
570  SElect rn
580  =1:di=left: =2:di=right: =3:di=up: =4:di=down
590  END SElect : i$=INKEY$(#1,wt)
600  :
610  REPeat loop
620  REMark flash_cursor
630  OVER -1: cd=0: a$="": b$="": try=try+1
640  pr dn,ac,'S': a$=INKEY$(#1,1)
650  pr dn,ac,'S': b$=INKEY$(#1,wt)
660  OVER 0: c$=a$&b$: cd=CODE(c$): no_ky=0
670  :
680  REMark Key determines snake direction:
690  SElect cd
700  =left : ac=ac-1: di=left : IF ac<1: lost
710  =right: ac=ac+1: di=right: IF ac>n: lost
720  =up   : dn=dn-1: di=up   : IF dn<1: lost
730  =down : dn=dn+1: di=down : IF dn>n: lost
740  =REMAINDER : no_ky=1
750  END SElect
760  :
770  REMark Direction when no key pressed:
780  IF no_ky THEN
790  SElect di
800  =left : ac=ac-1: IF ac<1: lost
810  =right: ac=ac+1: IF ac>n: lost
820  =up   : dn=dn-1: IF dn<1: lost
830  =down : dn=dn+1: IF dn>n: lost
840  END SElect
850  END IF
860  :
870  REMark New head position:
880  p=(dn-1)*n+ac
890  :
900  REMark you hit yourself:
910  IF g(p,3)=s OR g(p,3)=x: lost
```

Quanta Magazine

```
920 :
930 REMark The cell is empty:
940 IF g(p,3)=z THEN
950   g(p,1)=ac: g(p,2)=dn: g(p,3)=s: pr dn,ac,'S'
960   oac=q(kt,1): odn=q(kt,2): kt=kt+1
970   p2=(odn-1)*n+oac: g(p2,3)=z: pr odn,oac,' '
980   ct=ct+1: q(ct,1)=ac: q(ct,2)=dn
990 END IF
1000 :
1010 REMark Draw lengthened snake:
1020 IF g(p,3)=o THEN
1030   g(p,1)=ac: g(p,2)=dn: g(p,3)=s: pr dn,ac,'S'
1040   ct=ct+1: q(ct,1)=ac: q(ct,2)=dn
1050   cnt=cnt+1: IF cnt=rd_n: won
1060 END IF
1070 :
1080 END REPEAT loop
1090 ::
1100 :
1110 ::
1120 DEFINE PROCEDURE pr(dwn,acr,m$)
1130 AT dwn,acr: PRINT m$
1140 END DEFINE
1150 :
1160 DEFINE PROCEDURE lost
1170 INK 0: pr dn,ac,'*': INK 7
1180 AT 10,1: PRINT 'Game Over!try!'moves'
1190 BEEP 12345,67: epilogue
1200 END DEFINE
1210 :
1220 DEFINE PROCEDURE won
1230 AT 10,1: PRINT 'Bravo!try!'moves'
1240 BEEP 12345,6: epilogue
1250 END DEFINE
1260 :
1270 DEFINE PROCEDURE epilogue
1280 i$=INKEY$(#1,300): CLS
1290 pr 10,1,'Another Game? (y/n)'
1300 i$=INKEY$(#1,-1)
1310 IF i$='y': RUN: ELSE CSIZE 0,0
1320 WINDOW 256,206,256,0: STOP
1330 END DEFINE
1340 ::
```

End of Listing.

Quanta Magazine

TRADER'S CORNER

QUANTA recommends the following trading members to supply all your QL requirements, information and advice.

Trader's Advertisements can be viewed at:
<http://www.quanta.org.uk/traders>

RWAP SOFTWARE

3, Dale View Court, Fulford, Stoke-on-Trent, Staffordshire. ST11 9BA Tel: +44(0)1782 398143
[RWAP Software](#) provide a wide range of software and hardware for the Sinclair QL, Cambridge Z88 and ZX Spectrum computers.

Our Sinclair QL products include a range of all time classics to suit all users:

Adventures West, Return to Eden, Lost Kingdom of Zkul, The Prawn, Horrorday, Nemesis MKII

Games Qword, D-Day MKII, War in the East MKII, Grey Wolf, Open Golf, Stone Raider, Deathstrike, Hoverzone and Flightdeck

Other software

Q_Route - the famous route finder software for the QL. & QL Genealogist and Genealogy for windows.

Business Software Cash Trader, QL Payroll, Flashback SE
Image-D, QL Cosmos, ProForma ESC/P2 Drivers and Sideways

Programming Guides SBASIC/SuperBASIC Reference Manual on CD plus Q-Index & Q-Help

Replacement QL Keyboard Membranes

Second User Items probably the widest range of Hardware, Books and Software available anywhere

Plus Lots of similar items for the Cambridge Z88 and ZX Spectrum !!

Visit our Web Sites:

[RWAP SOFTWARE](#)

[RWAP ADVENTURES](#)

QLToday

ISSN 1432-5454

Now in Volume 12!

The Magazine about QL, QDOS, Sinclair Computers, SMSQ.....

German Office & Publisher:
Jochen Merz Software,
Kaiser-Wilhelm-Str.302
47169 Duisburg,
Germany

English Office:
QBranch
20, Locks Hill,
Portslade,
BN41 2LB
United Kingdom.

Tel: +49 (0)203 502011

Tel & Fax: +44 (0)1273 430501

Mobile: +44 7836 745501

Email: QLToday@J-M-S.com

Email: gbranch@gbranch.demon.co.uk

Visit our Web Page: <http://www.QLToday.com>

Quanta Magazine

JOCHEN MERZ SOFTWARE

Kaiser-Wilh.-Str 302, 47169 Duisburg, Germany
Tel: +49(0)203-502011 Fax: +49(0)203-502012
<http://SMSQ.J-M-S.com> SMSQ@J-M-S.com

SMSQ/E (QDOS compatible operating system), QPC (QL-Emulator for Pcs), QPCPrint (Epson printer driver emulator for Pcs) ... And all sorts of games, programmes and other application software and utilities for QDOS and SMSQ/E. We also produce the QL Today magazine. More details can be found on [our Web Site](#) (please see above) ...

All major credit cards, cheques, cash and PayPal accepted - plus foreign bank accounts in the UK, Austria, Switzerland, The Netherlands and USA to save money transfer costs!

Special Xmas Discounts

[See our Advert](#) or [Visit our Web Site](#) for full details

QBRANCH

20, Locks Hill, Portslade, Sussex. BN41 2LB
Tel & Fax: +44 (0)1273 430501

<http://www.qbranch.demon.co.uk> sales@qbranch.demon.co.uk

Feeling out on a limb? Reach out for Qbranch

Suppliers of computer solutions, Hardware and Software.

[See our Advert](#) or [Visit our Web Site](#) for full details

TF Services

29 Longfield Road, Tring, Herts, HP23 4DG.
Tel: 01442 828254 Fax/BBS: 01442 828255
<http://www.firshman.co.uk/ql> tony@firshman.co.uk

Manufacturer and supplier of QL accessories
Hermes, superHermes & superHermes LITE
RomDisq, Minerva, Mplane and I²C interfaces

Guaranteed QL repairs (UK only)

[See our Advert and pricelist](#) or [Visit us on line](#)

VISION MIXER 2 RELEASED

Dilwyn Jones

Vision Mixer 2 is now available from my website.

This is an updated version of my original Vision Mixer program of about 20 years ago. This release is for high colour modes (mode 32 and mode 33, and possibly mode 16, untested as yet on Aurora) systems only. The purpose is the same - screen picture slide shows with fancy visual effects for transitions from screen to screen. It can be used for presentations, slide shows, advertising display and so on. I use it, for example, to present a rolling slide show of information screens when I attend QL shows.

The program is pointer driven and needs high colour modes, Window Manager 2, plenty of expanded memory and Toolkit 2. All other extensions required (DJToolkit by Norman Dunbar, Display_Code extensions, QLiberator extensions, Easypr 4 extensions and Basconfig extensions are all linked into the QLiberator compiled program VM2_obj) so you need not worry about any of those.

The user interface has been completely rewritten using Easypr 4 to the extent that from the user's point of view this can be thought of as an original program, not an update, although the underlying concepts are broadly the same as the original Vision Mixer program from all those many years ago.

The fundamental improvements are:

- ❖ All new pointer driven, simplified user interface.
- ❖ Next screens always loaded from disk in real time during the viewing pause of the previous screens, frees up memory by not needing to hold all screens in memory, thus also allowing more pictures to be used in lists. Modern hard disk systems are pretty fast, so loading files in this way should not introduce much delay.
- ❖ Mixed screen mode screens may be used (mode 4, 8, 16, 32 or 33) with conversion done in real time at loading time, thus mode 4 screens may be used, with real time conversion to mode 32 done just before display using fast machine code routines from my BMP program.
- ❖ PIC files used, so that the program can determine picture size and mode number while loading, and so use just about any reasonable screen size.
- ❖ Not limited to 512x256 screen
- ❖ Can run in all screen resolutions - no direct writing to screen any more (uses pointer interface routines)

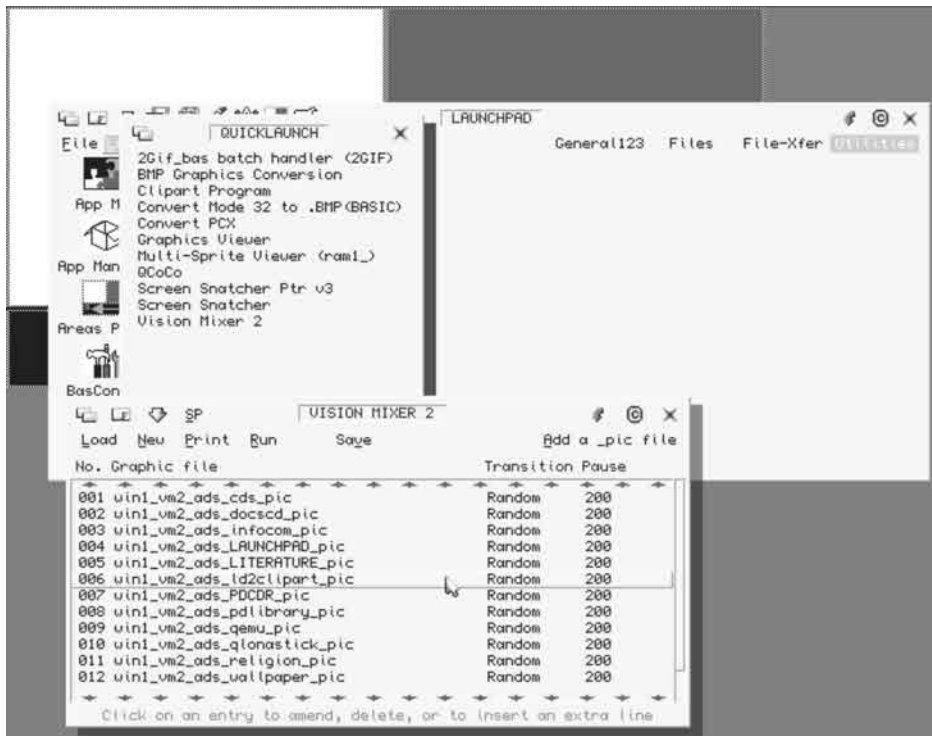
Quanta Magazine

- ❖ Does not have to use the whole screen - VM2 can use just part of a high resolution screen, allowing you to work on the rest of the screen and Vision Mixer can keep running.
- ❖ If your system allows background screen redrawing (e.g. recent QPC2), VM2 can continue to run if part obscured without destructive screen overwriting (as Vision Mixer 1 would do).
- ❖ Screen names list and visual effects list all combined into one control file for convenience.
- ❖ Preview window for video effects selected.
- ❖ Can use System Palette colour themes.

Vision Mixer 2 is available from the Graphics page on my website, a 66K download from:

www.dilwyn.uk6.net/graphics/index.html

Dilwyn Jones



Quanta Magazine

J - M - S NEWS

Jochen Merz

We are happy to offer you special Xmas discounts on many of our products, up to 30% off the regular price. Please visit SMSQ.J-M-S.COM or have a look at our ad on the Quanta website.

We are happy to be able to provide our valued customers with dedicated accounts for many countries like the UK (£ sterling and EUR account), Austria, Switzerland (CHF and EUR account), The Netherlands and the USA.

We are also able to accept UK £ sterling cheques and US\$ cheques from the USA. Of course, all the other payment methods like Visa, Mastercard, Diners Club or PayPal remain.

[I make no apology for filling the second half of the magazine with some more of Steve Poole's programmes. I hope that you can find the time to type some of the listings into your QL (or emulator) and letting the children (or Grandchildren!) have a go. Well, it's that time of year. ENJOY - Ed]

MAD POLYGONS

Steve Poole

In a former Quanta magazine, Al Boehm showed us a routine for drawing fractal crosses. A surprising and stunning result. In a science magazine dated May 1996 dealing with topology, there is a written account of an algorithm to draw branched polygons. Having deduced how to code this in SuperBASIC, I shall let you discover the output which needs no comment. The guy that discovered this must have been surprised, and if superstitious, spooked out!

Best Wishes,

Steve Poole

Quanta Magazine

```
100 ::
110 REMark polyGONEmad_bas
120 REMark by S.Poole, v25nov07
130 :
140 CLS: SCALE 20,-9,-9: PENDOWN
150 FOR angle=72,90,120
160   right=angle: left=-angle
170   POINT 0,0: TURNT0 0
180   FOR sides=1 TO 5
190     MOVE 1: TURN left
200     MOVE 3: TURN left
210     MOVE 2: TURN right
220     MOVE 1: TURN right
230     MOVE 2: TURN left
240   END FOR sides
250   PAUSE 150: CLS
260 END FOR angle
270 ::
```

End of Listing.

PRIORITIES

Steve Poole

In Quanta vol 24, issue 1 of March 2007 I discussed parentheses. Line 110 of listing two did not work and was included, (of course), to illustrate what good SuperBASIC statements should not resemble.

This time I will explain one of the main problems with priorities, and I promise that the code is guaranteed...

When running code, the SuperBASIC Interpreter first tries to unravel any statements following a PRINT keyword, reading from left to right. If it finds sub-expressions it will evaluate them in the order of any brackets, then inside these in strict arithmetical order, or in case of equal precedence, from left to right. Then it will back-track up and down its path of priorities, altering any global variables assigned to actual parameters as it goes. Finally, all global variables kept in hand will be printed from the left-most towards the right.

Here is an example taken from the listing:

```
180 n=0: PRINT func_1(n),n \ func_2(n),n
```

Quanta Magazine

Run the program and look at the results printed out for Lines 140 and 160. These are perfectly logical. But what is happening with Line 180? The result desired should be 1 10; 2 110 but the global variable n is reassigned first by func_1, then by func_2, (before the interpreter prints the first value of n), so when this is printed it has already been re-assigned by func_2!

This sort of unwanted reassignment happens a lot if you nest procedures and functions deeply inside many levels of brackets, or within multiple print statements, where they may all interfere with each other. I could give plenty of other really horrible examples, but that would only add to the confusion!

Whatever you do, remember, SuperBASIC is a very flexible language allowing the construction of very complex statements. But especially when writing prototype code, always keep things simple, because debugging can otherwise be a nightmare when there are these obscure reassignments occurring. The solution to the line 180 bug is therefore to split the Print statement over two lines. Simple, elegant and hassle-free!

This article was written to prolong and develop the previous 'parentheses' one and the 'LOCal' article prepared for another issue of Quanta, as these subjects comprise some of the worst programming traps on the QL... and ones that are seldom, if ever, referred to in QL literature.

Best Wishes,

```
100 ::  
110 REMark Priority_bas, by S,Poole. v14mar2007  
120 CLEAR: CLS  
130 :  
140 n=0: PRINT func_1(n),n\  
150 :  
160 n=0: PRINT func_2(n),n\  
170 :  
180 n=0: PRINT func_1(n),n\  
190 :  
200 DEFine FuNction func_1(p)  
210 p=p+10: RETurn 1  
220 END DEFine  
230 :  
240 DEFine FuNction func_2(p)  
250 p=p+100: RETurn 2  
260 END DEFine  
270 ::
```

End of Listing.

A favourite party-game when I was young was called 'At the Market'. There could be as many players as you liked. Here's how it plays :

The first person says 'At the Market I got an Apple.' The second continues 'At the Market I got an Apple and a Pear.' The game continues with each player adding one object to the end of the list. The problem for me, as a program-writer, was that a QL program to play this game would require a QL Dictionary, which not all readers possess.

So, instead, this routine strings out mere numbers and the QL checks to see if any memorisation errors have been made. The winner is the player with the highest score! This game is very good for training human memory, and I seem to remember getting scores of around 35 whilst playing 'At the Market' with friends when young... perhaps using real objects helps stimulate the brain?

As the strings get longer, the program prolongs the pauses, but you can vary these using the Difficulty-Level variable : '1' is the easiest. If you wish, try modifying the program for a larger number of players, with the QL eliminating those that give a wrong answer until only one player remains...

Best Wishes,

```
100 ::
110 REMark Repeat_bas. by S.Poole, v8sept2006
120 REMark (for Quanta)
130 OPEN#1,con_128: WINDOW 512,256,0,0: CLS
140 high=0: start: d=99
150 :
160 REPEAT loop
170 m$=m$&RND(0 TO 9)
180 AT 3,1: PRINT!'Ready...!m$
190 i$=INKEY$(#1,d): AT 3,1: CLS 3
200 AT 5,1: CLS 3: INPUT!'Your Series?!s$
210 :
220 IF s$<>m$ THEN
230 AT 7,1: PRINT!'Game Over!'
240 AT 9,1: PRINT!'Your Score!'ct
250 AT 11,1: PRINT!'Top Score:'!high
260 AT 13,1: INPUT!'Another Game? (y/n)!g$
```


Quanta Magazine

```
270 IF g$="": GO TO 260
280 IF g$ INSTR'nN': EXIT loop
290 start: NEXT loop
300 END IF
310 :
320 AT 5,1: CLS 3: PRINT 'Correct!',,
330 i$=INKEY$(#1,50): AT 5,1: CLS 3
340 ct=ct+1: IF high<ct: high=ct
350 END REPEAT loop: WINDOW 256,206,0,0: STOP
360 :
370 DEFine PROCedure start
380 CLS: AT 1,1: INPUT'Difficulty Level? (1 to 3)'!d$
390 IF d$="": GO TO 380
400 IF d$ INSTR '123': ELSE GO TO 380
410 m$="": ct=1: d=(150*ct)/d$
420 END DEFine
430 ::
```

End of Listing.

STADDON

Steve Poole

First of all I must apologise to Quanta members for having messed up last Xmas' competition. First, one of the formula for calculating trajectories printed was erroneous. (It was false too in the physics text-book I got it from). I noticed this and modified the clue-program to fix the error, but alas! changed the wrong line! So to make amends I decided to write a whole trajectory-based program for the magazine, which, unlike the above code, has been beta-tested. This game should be especially good for those whose notions of Physics need revising.

The game involves two enemies trying to destroy each other with rockets. To make the job harder, there is a high wall somewhere between them, but they have spotters to indicate their accuracy. So the enemy uses your trajectories to home in on you!

The difficulty-levels determine the size of the targets and the height of the wall. To play, you have to input the estimated velocity and angle required to hit the enemy, (in this case the QL). To trace your progress, when the game is over, the number of shots you fired is indicated. You should be able to discover a winning strategy after a while. If the noise annoys, LET BP=0...

Quanta Magazine

The name of the program is called 'Staddon' after the huge defensive wall built on Staddon Heights, (near Plymouth), to prevent enemy ships being able to shell the Dockyard. This wall is a major landmark, and one which has made many tourists try to guess what on earth it was built for.

In real-life situations, every time you fire a ballistic weapon you hasten to move it a short distance to confuse your assailants. Of course the enemy does the same thing, so to improve on the program, try adding ducking and weaving behaviour to both parties. That would complicate the QLs homing-in performance somewhat. (But I have already done it and it is quite easily done).

The flow-chart for this program is rather confusing, as the code contains loop-independent fall-throughs using flags to trigger the loop epilogues. This is to allow both opponents to fire in the same loop, and also to erase the vapour-trails. It is best to print the listing before trying to understand it. Be careful not to modify the random ranges, as these take a lot of experimenting to get right, and take as much time to determine as writing the whole of the code!

No doubt this program could be optimised, but it only took me three sessions to conceive and write it, and having been beta-tested, at least it works properly. If I get the time, I might try to rewrite it to make it shorter...

Happy Bombardiering!

Best Wishes.

```
100 ::
110 REMark Staddon_bas, by S.Poole, v11feb2007
120 REMark for Quanta, Beta-test by B.Coativy.
130 :
140 REMark Initialise the variables:
150 CLEAR: OPEN#1,con_16: MODE 4: WINDOW 512,256,0,0: CLS
160 BORDER 1,2: SCALE 256,0,-10
170 AT 0,1: INPUT'Difficulty 1 to 9?'!dif$
180 IF dif$="": GO TO 170
190 IF dif$ INSTR '123456789': dif=dif$: ELSE GO TO 170
200 g=9.81: rds=12-dif: stp=5E-2: ps=1: bp=1
210 SElect dif
220 =1 TO 3:fc=.98: =4 TO 6:fc=.99: =7 TO 9:fc=.995
230 END SElect
240 :
```

Quanta Magazine

```
250 REMark Randomise targets and wall:
260 HX=RND(5 TO 181) : HY=0
270 KX=RND(191 TO 373): KY=0
280 Wx=RND(HX+25 TO KX-25): ww=5*dif
290 wz=15*dif: wy=RND(ww TO wz)
300 odx=KX-HX: try=0
310 s1x=KX-rds: s2x=KX+rds
320 w1x=Wx-rds: w2x=Wx+rds
330 h1x=HX-rds: h2x=HX+rds
340 :
350 INK 3: CIRCLE HX,HY,rds ; KX,KY,rds: INK 4
360 LINE 0,0 TO 382,0: FILL 1: INK 5
370 CURSOR HX,HY,-2,-4: PRINT 'H'
380 CURSOR KX,KY,-2,-4: PRINT 'Q': INK 3: FILL 1
390 LINE w1x,0 TO w2x,0 TO w2x,wy TO w1x,wy TO w1x,0
400 FILL 0: INK 6
410 :
420 REPEAT loop
430   try=try+1
440   :
450   REMark Human chooses range details:
460   AT 1,1: CLS 3: INPUT 'velocity (1 to 127):'!v$
470   IF v$="": GO TO 460
480   IF v$>0 AND v$<129: v=v$: ELSE GO TO 460
490   AT 2,1: CLS 3: INPUT 'angle 1 to 90):'!a$
500   IF a$="": GO TO 490
510   IF a$>0 AND a$<91: a=RAD(a$): ELSE GO TO 490
520   :
530   REMark Draw the vapour trail:
540   OVER -1
550   FOR f=1 TO 2
560     INK 1: LINE HX,HY: ohy=HY: oqy=QY
570     :
580     FOR t=0 TO 32767 STEP stp
590       x=v*COS(a)*t: ohy=y
600       y=v*SIN(a)*t-.5*g*t*t
610       phx=HX+x: phy=HY+y
620       LINE TO phx,phy: i$=INKEY$(#1,ps)
630       IF f=1: IF bp: BEEP 12,t+55
640       :
650       REMark Did it hit the ground:
660       IF y<0: EXIT t
670       :
680       REMark Did you hit the wall:
690       IF phx>w1x AND phx<w2x THEN
```

Quanta Magazine

```
700     IF phy<wy: CIRCLE phx,phy,rds: EXIT t
710     END IF
720     :
730     END FOR t
740     :
750     REMark Did you blast yourself:
760     IF f=2 THEN
770         IF phx>h1x AND phx<h2x THEN
780             BEEP 12345,67: INK 7
790             PRINT' You shot yourself in!'try!'shots'
800             i$=INKEY$(#1,250): RUN
810         END IF
820     END IF
830     :
840     REMark Draw the explosion:
850     IF f=1: BEEP 12345,255
860     FILL 1: CIRCLE phx,phy,rds: FILL 0
870     :
880     REMark retain final target x coordinate:
890     IF f=1: Dx=INT(x): d1x=Dx-rds: d2x=Dx+rds
900     END FOR f: OVER 0
910     :
920     REMark Did you hit the QI:
930     SElect odx
940     =d1x TO d2x
950     BEEP 12345,6: AT 4,1: INK 7
960     PRINT'You won in!'try!'shots'
970     PRINT' Another? (y/n)'
980     i$=INKEY$(#1,-1)
990     IF i$=='y': RUN: ELSE MODE 4: STOP
1000    END SElect
1010 :
1020 :
1030    REMark QL homes in on you:
1040    IF odx>Dx: a=a*fc: v=v/fc
1050    IF odx<Dx: a=a/fc: v=v*fc
1060    :
1070    REMark QL fires:
1080    OVER -1
1090    FOR f=1 TO 2
1100        INK 6: LINE KX,KY
1110        :
1120        FOR t=0 TO 32767 STEP stp
1130            x=v*COS(a)*t: oqy=y: y=v*SIN(a)*t-.5*g*t*t
1140            pqx=KX-x: pqy=KY+y
```

Quanta Magazine

```
1150     LINE TO pqx,pqy: i$=INKEY$(#1,ps)
1160     IF f=1: IF bp: BEEP 12,t+55
1170     :
1180     REMark Was the ground hit:
1190     IF y<0: EXIT t
1200     :
1210     REMark Was the wall hit:
1220     IF pqx>w1x AND pqx<w2x THEN
1230         IF pqy<wy: CIRCLE pqx,pqy,rds: EXIT t
1240     END IF
1250     :
1260     END FOR t
1270     :
1280     REMark Did the QL blast itself:
1290     IF f=2 THEN
1300         IF pqx>s1x AND pqx<s2x THEN
1310             BEEP 12345,255: INK 7
1320             PRINT' You won in!'try!'shot(s)'
1330             i$=INKEY$(#1,250): RUN
1340         END IF
1350     END IF
1360     :
1370     REMark Draw the explosion:
1380     IF f=1: BEEP 12345,255
1390     FILL 1: CIRCLE pqx,pqy,rds: FILL 0
1400     :
1410     REMark See if QL won:
1420     IF f=1: Dx=INT(x): d1x=Dx-rds: d2x=Dx+rds
1430     END FOR f: OVER 0
1440     :
1450     REMark Did you hit the human?
1460     SElect odx
1470         =d1x TO d2x
1480         BEEP 12345,25: INK 7: AT 4,1
1490         PRINT'QL Won in!'try!'shot(s)'
1500         PRINT' Another? (y/n)'
1510         i$=INKEY$(#1,-1): IF i$=='y': RUN: ELSE STOP
1520     END SElect
1530     :
1540     END REpeat loop
1550 ::
```

End of listing.

Otherwise known as 'Noughts and Crosses', this was one of the very first games to be subjected to the assault of Artificial Intelligence Programmers. It proved to be amazingly difficult to get a computer to play well against a human opponent, as the amount of coding involved is far greater than one might imagine.

Even finding a winner demands sorting through 8 out of 23 combinations... and if the computer plays, it must foresee some 362880 (Factorial 9!) different possible moves. (There are not only 'X' & 'O' on the grid, but also empty spaces). You don't believe me? ... Then just try playing every possible game and writing out a grid at every stage! Even though we can reduce this figure taking into account obvious symmetries, that's still one heck of a lot of decisions for an AI program to make.

The usual method of making decisions is to create a 'binary tree' as you go along, with the computer looking several steps ahead at every branch, using a data-base of all possible moves held in memory. This is quite clearly a remarkable example of how little we tend to appreciate the astounding power of our own brains. I have deliberately constructed this program with, literally, a 'DATA'-base of winning positions, to reduce the amount of coding necessary to just one FOR loop, 'J', which has only to search the latest INPUT position, and not all possible combinations. That is, the string 'q\$' contains the number of possible winning combinations per grid-figure. The array 'N\$()' holds the actual winning triples per grid-figure. Therefore, the QL has only to check between 2 and 4 triples, not 23, per digit and certainly not the 288 outcomes, nor the 362880 possible games-moves! Yet this game is relatively simple : there are just nine positions on the grid : For chess with 64 positions there are 10^{40} nodes, needing 10^{21} years to compute on a 128k QL!

This program essentially imposes 'bottom-up' checking. Starting from the base of the combinations-Pyramid, we work upwards to the summit in one go. This means that, for example, if there were 65536 outcomes, we would only need to lookup 16 nodes! Essentially, this is one approach to parallel processing : And being how the brain operates, by combining all possible outcomes, and then working back up to the top of the neural tree in just a couple of dozen steps, and that almost instantly! Of course this strategy only works serially all the same on one unique processor, but would be easy to program on a super multi-processor computer.

Quanta Magazine

There is no point in describing the program further here. It is sufficiently REMark_ed, and the front-end is intuitive if you have already played the game on paper. All you need to decide is which player is O and who is X.

Tic-Tac-Toe seems to us to be almost obvious to play. I have started writing an AI solver for this program. If I ever get round to finishing it, I will submit it in Quanta. After that the QL could always play itself : At least then it would always win! If anyone knows of a simple solution, then they could contact the Editor and let us all into the secret. If you are really keen, try getting the QL to play against a human yourself. Even AI books don't hint at the appropriate method, other than using randomly-generated guesses in a self-learning structure! The only condition is that the code must be short and compact : A ten-thousand line COBOL solution would not be considered as suitable...

Enjoy your 'OXO'!

Best Wishes,

```
100 ::
110 REMark TicTacToe_bas, by S,Poole, v22sept2006
120 :
130 CLEAR: RESTORE : REMark for Quanta
140 OPEN#1,con_32: WINDOW 512,256,0,0: CLS
150 WINDOW 256,206,256,0: CSIZE 3,1
160 DIM n$(9,4,3): q$='323242323': t$='123456789'
170 d=RND(1 TO 2): SElect d: =1:d$='OX': =2:d$='XO'
180 :
190 REMark Read the triples related to each f$-digit:
200 FOR f=1 TO 9
210   READ f$: FOR j=1 TO q$(f): READ n$(f,j,1 TO 3)
220 END FOR f: n=1
230 :
240 DATA 1,123,147,159, 2,123,258, 3,123,369,357, 4,147,456
250 DATA 5,258,456,159,357, 6,456,369, 7,147,789,357
260 DATA 8,789,258, 9,789,369,159
270 :
280 REMark Print the display:
290 FOR y=1 TO 3
300   FOR x=1 TO 3: AT y,x: PRINT n: n=n+1
310 END FOR y: kt=1
320 :
```

Quanta Magazine

```
330 REPEAT game
340   REMark For each Player:
350   FOR f=1 TO 2
360     REMark Choose a digit in grid:
370     f$=d$(f): Cue f$: i$=INKEY$(#1,-1)
380     IF i$="": GO TO 370
390     IF i$ INSTR t$: ELSE GO TO 370
400     :
410     REMark Get number of triples per digit:
420     i=i$: Qsi=q$(i): Print_em i$
430     REMark Inscribe chosen letter in t$:
440     t$(i)=f$
450     :
460     REMark scan all aligned triples:
470     FOR j=1 TO Qsi
480       k$=n$(i,j,1 TO 3): ct=0
490       :
500       REMark See if all O's or all X's:
510       FOR k=1 TO 3
520         IF t$(k$(k))=f$: ct=ct+1
530         IF ct=3: EXIT game
540       END FOR k
550     END FOR j
560     kt=kt+1: IF kt=10: more 'DRAW!'
570   END FOR f
580 END REPEAT game
590 :
600 FOR s=1 TO 3: m$=f$: f$="*": Print_em k$(s): f$=m$
610 AT 7,1: PRINT f$: more 'WON!'
620 ::
630 DEFine PROCEDURE more(me$)
640 AT 5,0: CLS 3: AT 6,0: CLS 3
650 AT 7,3: PRINT me$
660 AT 8,1: PRINT 'Another? (y/n)'
670 q$=INKEY$(#1,-1)
680 IF q$="y": RUN: ELSE CSIZE 0,0: CLS: STOP
690 END DEFine
700 :
710 DEFine PROCEDURE Print_em(p$)
720 p=p$-1: xx=p MOD 3: yy=p DIV 3
730 AT yy+1,xx+1: PRINT f$
740 END DEFine
```


Quanta Magazine

750 :
760 DEFine PROCedure Cue(c\$)
770 AT 5,1: CLS 3: PRINT c\$!'to play...'
780 AT 6,1: CLS 3: PRINT'Hit a Digit...'
790 END DEFine
800 ::

End of Listing.

SUBGROUP MEETINGS

THE WEST MIDLANDS QL USER GROUP

This long-running group has moved back to its previous venue, the Queens Head pub in the City Centre. Meetings remain open to Quanta members and non-members alike. The Queens Head is in Steelhouse Lane, Central Birmingham, four minutes walk from Snow Hill main-line railway station and has ample street parking nearby.

Meetings are usually on the first and third Monday of each month (but not on Bank holidays unless the previous one was cancelled for that reason). After your first meeting you're expected to pay £1 each time towards the organisation and monthly newsletter costs - except once a year when the group pays all those who attend a pound instead (We provide food twice a year for those who attend the AGM and the group's birthday party) and at meetings we discuss all sorts of things, QL-related and otherwise, usually from about 8:20pm onwards till 10:30 or later. The group owns QL hardware, a disk library and subscribes to relevant magazines. People can join the mailing list for £3.50 per year, even if they can't attend regularly.

The contact number is Mike Bedford White's, on 0121 605 4438, anytime after 11 am.

Email: mikebwhite@blueyonder.co.uk

QUANTA DORSET SUBGROUP

Meetings are held, on the second Sunday of each month at the Merley Community Centre in Harrier Drive, Merley, just south of Wimborne Minster. Time: 2.00pm to 5.00pm. All are welcome.

John Meadows Tel: 01202 576189, or John Mason, 01425 275894.

Quanta Magazine

LONDON QL AND QUANTA GROUP

Meetings are held in the School Room, which is the basement of the Borough Welsh Congregational Chapel, 90, Southwark Bridge Road, London SE1. This is almost opposite the junction with Marshalsea Road, at the other end of which, 5 minutes walk away, is the Borough Underground Station. Free parking is easily available in Southwark Bridge Road for road users, Cyclists are welcome.

Members currently each pay a subscription of £30 per year towards the cost of the hire of the hall and a small additional charge is made for tea/coffee - biscuits are then provided free.

Please bring your QL equipment if possible, all types welcome, e.g., PC portable running emulators. The Group has some systems stored at the venue for its use. In addition, there are lots of donated second user equipment, books, magazines, etc, available for purchase.

Time: 2.00pm to 6.00pm, on the second Sunday of the month.

No meetings are held in either December - winter break, or August summer break. Giving 10 meetings a year.

Malcolm Cadman: Tel: 020 8691 5780

email: QL@mcad.demon.co.uk

NEMQLUG - THE NORTH EAST MANCHESTER QL USER GROUP

Meetings are held at 181, Urmston Lane, Stretford, Manchester. M32 9EH (Near Junction 7 of M60 Motorway - Was Junction 7 M63 Motorway). Our informal meetings start at 7 p.m. and finish at 11p.m. All welcome.

Meets on the last Thursday of each month (except December).

John/Sarah Gilpin 0161-865 2872 (treasurer@quanta.org.uk)

Quanta Magazine

SCOTTISH QL USERS GROUP

SQLUG meets at George Gwilt's home every second Sunday in the month. Please contact George Gwilt before coming to get the location.

Geogwilt@aol.com

SOLENT SUBGROUP

Meets at Botley Market Hall, all welcome. Park at the back and use side entrance.

1st Saturday of the month from 2.00 pm to 6.00 pm.

Graham Evans, Tel: 023 8040 3350.

SURREY QUANTA SUBGROUP (SQSG)

Venue is St Giles Church Hall, Ashted. (This is the other side, towards the downs, of the main Epsom - Leatherhead road, up Park Lane which is at the Epsom end of the town).

From 8.00 to 10.30 pm on the last Wednesday of each month, except December.

Ken Bain 01932 347 432 (to midnight), kenb@bcs.org.uk

SUSSEX QL USER GROUP

Currently without a meeting place, anyone interested should contact Keith Mitchell or Roy Wood.

Roy Wood, 01273 430501 or Keith Mitchell, 01903 742263.

